

Foreword

The area of dynamic and on-line algorithms has generated a rich stream of theoretically interesting results with a wide variety of applications in relevant fields of Computer Science. Indeed, in several advanced computer applications, both the data and the operations to be performed on these data are only partially known and vary with time. Typical situations of this kind arise in many areas, such as real-time manufacturing systems, man–machine interfaces, robotics, scene analysis, computer graphics, and communication networks.

One obvious way of tackling this kind of problems would be to recompute their solution from scratch after each input change. Unfortunately, this is often computationally expensive. More efficient approaches try to retain some information between subsequent updates so as to react quickly in response to input changes. This has generated many new algorithms and data structures for solving efficiently dynamic graph problems, where some graph property must be maintained while the graph is undergoing a sequence of updates such as edge insertions, deletions and cost changes. Some efforts to analyze the concept of incremental computation from an abstract point of view have also been undertaken.

Another approach that has been developed tries to minimize the total cost of the operations to be processed, so that this cost is not much worse than the cost of the *off-line* algorithm (i.e., the algorithm that has complete knowledge about the sequence of input changes). Since the seminal papers of Sleator and Tarjan in which this type of problems was first investigated, various research directions have been developed. Several researchers have studied on-line solutions of combinatorial problems often arising in the context of parallel and distributed system resource management, such as job scheduling, servers management, routing, etc. For these problems efficient algorithms have been designed for the case in which the sequence of operations to be performed is not known in advance and their competitiveness with respect to the best off-line algorithms is evaluated.

This special issue contains papers that cover a broad range of current research in this field. The papers were selected among a large number of manuscripts received in response to a call for papers. Due to the high quality of the submissions, some of the manuscripts that were not selected will appear later in regular issues of this journal. The list of topics covered by the selected papers include on-line scheduling, competitive algorithms, incremental computation, dynamic distributed algorithms and their lower bounds, and dynamic graph algorithms.

In “On-line Scheduling of Jobs with Fixed Start and End Times”, by Gerhard Wöginger, the problem of scheduling a set of jobs on-line to a single machine is considered. Each job has a value associated with it, and it has fixed start and end times, namely it must be processed in a given time interval. Jobs not processed immediately after their arrival, or jobs whose processing is interrupted are lost. The goal is to maximize the total value of all processed (and not interrupted) jobs. In general, this problem does not allow on-line approximations with finite worst-case guarantee. Matching upper and lower bounds for classes of approximations algorithms are given.

In “Non Clairvoyant Scheduling”, Rajeev Motwani, Steven Phillips and Eric Torng give competitive algorithms for pre-emptive scheduling when jobs arrive with no information as to their length. The main goal is to minimize the total idle time. The algorithms presented are relevant to both the theory and the systems communities.

In “Dynamic Scheduling on Parallel Machines”, Anja Feldmann, Jiří Sgall and Shang-Hua Teng investigate on-line scheduling on a range of parallel machines from the PRAM to fixed connection networks such as grids and hypercubes. The problem is treated in a rather general fashion, and matching upper and lower bounds are obtained for most of these parallel architectures.

The paper “On-line Load Balancing”, by Yossi Azar, Andrei Z. Broder and Anna R. Karlin, considers the scenario where there are n servers that must complete a sequence of tasks, and such that each task can be handled only by a subset of the n servers. The goal is to assign each task to an appropriate server so as to minimize the maximum load on the servers. Matching upper and lower bounds are derived for the competitive ratio of on-line greedy algorithms.

In “Competitive Algorithms for the Weighted Server Problem”, Amos Fiat and Moty Ricklin propose an interesting variation of the k -server problem, where each server is assigned a weight, and the cost incurred in moving a server is the weight of the server times the distance moved. This problem has a nice application to storage management, where the write times for different types of memories differ. It is shown that the best competitive ratio that can be achieved by a deterministic algorithm is exponential in k , even in the case where each server has one of two possible weights. In this special case, however, a randomized algorithm with an $\tilde{O}(k^3)$ competitive ratio is given.

In “The List Update Problem and the Retrieval of Sets”, Fabrizio d’Amore and Vincenzo Liberatore generalize the list update problem by allowing the retrieval of

sets, and give both competitive algorithms and lower bounds for this problem. This is a step towards the analysis of structures more complex than linear lists, for which competitive algorithms are known only in particular settings.

In “Constructing Competitive Tours from Local Information”, Bala Kalyanasundaram and Kirk R. Pruhs show how to traverse the vertices of a weighted planar graph on-line. Initially, the searcher has no knowledge on the topology and the weights of G . Whenever he/she arrives at a vertex, the weights and endpoints of all edges emanating from this vertex are revealed to the searcher. The searcher’s goal is to visit each vertex of G , through a path which is as short as possible. A constant competitive algorithm for this problem is presented.

In “Fully Dynamic 2-Edge Connectivity in Planar Graphs”, John Hershberger, Monika Rauch and Subhash Suri present a data structure for maintaining 2-edge connectivity information dynamically in an embedded planar graph. The data structure is based upon the topology tree of Frederickson, and introduces two new concepts: edge bundles and coverage graph recipes. The former is a method for collapsing and manipulating edges that belong to the same equivalence class for a given partition of vertices. The latter is a method for compressing and uncompressing portions of a planar graph. Using these two new ideas yields time bounds of $O(\log^2 n)$ per update and $O(\log n)$ per query.

In “Lower Bounds for On-Line Graph Coloring”, by Magnús M. Halldórsson and Mario Szegedy, on-line graph coloring is considered. The vertices of a graph are assumed to arrive one-by-one, along with their edges to the already arrived vertices; as soon as a vertex arrives, it must be given a legal, irrevocable color, with no knowledge of the future. The objective is to find a small competitive ratio, with respect to the number of colors used by the off-line algorithm (i.e., had the algorithm known the future). Lovász, Saks and Trotter presented an algorithm with performance guarantee $O(n/\log^* n)$. This paper presents a pretty close lower bound of $\Omega(n/\log^2 n)$ on the possible competitive ratio, even for randomized algorithms with a fairly weak (oblivious) adversary.

In “Lower Bounds on the Competitive Ratio for Mobile User Tracking and Distributed Job Scheduling”, Noga Alon, Gil Kalai, Moty Ricklin and Larry Stockmeyer prove lower bounds for two related problems in important areas of distributed computing. These lower bound use elegant arguments for high-girth graphs and expanders, as well as harmonic-analytic inequalities in a novel way.

In “Complexity Models for Incremental Computation”, by Peter Bro Miltersen, Sairam Subramanian, Jeffrey Scott Vitter and Roberto Tamassia, a complexity theoretic approach to incremental computation is undertaken. The authors define the notion of incremental complexity, introduce incremental complexity classes, and prove several results in this model, including some completeness theorems.

It is our hope that this special issue generates more interest and cross-fertilization in the areas of dynamic and on-line algorithms, and that it will help in bridging the gap between theoretical and practical algorithms. We would like to thank the authors for submitting papers of exceptionally high quality, and the referees for their carefully

reviewing the manuscripts in a timely manner. We would also like to thank the Editor-in-Chief Maurice Nivat for making this special issue possible.

GIORGIO AUSIELLO AND GIUSEPPE F. ITALIANO
Guest Editors